# EXHIBIT 140-B

## Redacted Version of

## Document Sought to be Sealed

**From:** Jackie Cerretani Frank </O=THEFACEBOOK/OU=EXCHANGE ADMINISTRATIVE GROUP (FYDIBOHF23SPDLT)/CN=RECIPIENTS/CN=JCERRETANI>
**Sent:** Monday, November 30, 2009 3:04 PM
**To:** Reply Handler
**Subject:** Re: GDP master thread

Jackie Cerretani Frank - at 3:04pm
Sorry that is so unreadable -- I can't get on the VPN from home today, so had to send by email. Will re-post a formatted version tomorrow. In the mentime, I'll email it to you guys in a seperate message.
Jackie

Comment on the discussion at:
http://www.intern.facebook.com/intern/discussions/?fbid=148170304435

Reply to this email to post a comment. To unsubscribe, reply to this email with the word 'unsubscribe'.

---

On 10/13/09 18:29:28, Mark Kinsey wrote:

Title: GDP master thread
Tags: platform, ideas
Sent To: Julie Zhuo, Kelly Winters, Ruchi Sanghvi, Jackie Cerretani Frank, Yariv Sadan by Mark Kinsey

Ok. Here's where we are:

Ancient wiki from Ruchi: http://www.intern.facebook.com/intern/wiki/index.php/Platform/auth_model

The initial mini-plan: http://www.intern.facebook.com/intern/wiki/index.php/Platform/granular

Open questions with suggestion/plan:

1) What to do about friend data?
- Is the app an agent of the user, OR is it totally separate
- How many apps really need friend data
- Is asking a user to grant access to friend data too awkward (or just right!)
- Will floodgate ensure enough everyone data that the point is moot

PLAN:
- Kelly gets data/examples of where friend data is necessary for apps (DO NOW)
- Mark/Yariv try and run data from API about friend data usage (DO NOW)
- Mark checks with Ruchi about floodgate expectations on % everyone data (DO NOW)

2) Storage vs usage
- Do apps get to store the user data? How does this fit with other plans going on in zuck's head?

- Do apps get to store whatever friend data they get?

SUGGESTION:
- Apps can store any data they get. They have an incentive to keep it fresh

3) UI/UEX (dependent upon 1)
- Number of screens
- Icons/no icons
- Just use "Do you trust this app"-type language?

PLAN:
- Get some basic data on 1), make a few mocks, and test with users

4) Migration plan
- Do we map old settings to new ones OR to apps have to re-request data using new flow?
- How much time to give?

SUGGESTION:
- Announce and launch new flow on Day X, tell developers they have 60 days to request all required data (old data IS NOT mapped). Basically, they can switch to new flow at any point
- 60 days later, stop using TOS.php flow altogether. Any unprepared app will probably break, any apps relying on old permissions that didn't use the new flow will probably break

5) Getting it out there asap so we can get real data
- Why not just create a prototype and invite/force a few apps to run it, monitor for user behavior, and iterate?

SUGGESTION:
- I like this idea. Developers might well want to be involved because it would skip TOS. Not sure how this fits with floodgate yet (as in, what data do they automatically get)

--------------------
Full History
--------------------

Kelly Winters - at 5:22pm on October 14th Hey guys,

Here's the feedback I received: https://docs.google.com/a/facebook.com/Doc?id=dg62k3kb_7nghk95d5

It seems to break down into three categories:
1. "Not that private" data that we'd be sadder to lose access to 2. "More private" data that we'd rather users opt-in to share with an app 3. Aggregators/desktop viewers

Use cases:
* "Not that private data"
  o No inviting based on stated interests (e.g., other friends that like chess)
  o Tell me who is having a good/bad day based on their status message
  o No more collaborative filtering (e.g., recommended books/music based on my friends' stated preferences)
  o No more online presence
    + Apps would not be able to tell a user that their friend is available to get engaged to use the app at the same time
* "More private data"
  o No more dating apps (no relationship status)
  o No rich experiences based on detailed user data
    + Load viewing history of my friends' TV watching habits, recommend TV shows
* "Lost aggregated viewing functionality"
  o No more content viewers/aggregators (notes, photos, etc.)
  o Would this kill all desktop apps?

**reply sent via email**

Mark Kinsey - at 9:12pm on October 21st
Ruchi tells me they hope to get 60-70% everyone data - which make a big difference to us. All of these negatives only happens when the friend data is not public.

Still need to run data from API though I'm not sure how useful it will be (because I can't think how to know is user.getInfo is getting called with user=me or user=friend of me) Perhaps fql.mulitquery will give us some clues. Or maybe we could assume if > 1 id is passed that it's for friend at (END OF WEEK FOR THIS)

We should have some idea of our roadmap for this. Our launch will be gated on floodgate, but I think we only need to worry about: 1) the % of everyone data, 2) the buckets to use. We can continue without knowing these things. So I suggest:

Week of:
this - get all friend data required
26th - dialog options mocked
31st - user testing
7th - tie it all together

then wait for floodgate developments

Mike Vernal - at 4:51pm on October 22nd
Notes:

1/ I like Read / Write permissions, but I don't think all permissions will fall into this bucket. We may have things like "Bookmark," "Offline Access," etc. that we should treat in a consistent way. So maybe read, write, other or something?

2/ How do you build Facebook if you can only access your friends' everyone data? Why can't you access all everyone data in the system? I think we could do something where we rate limit the data a single session can get to make sure people aren't crawling the graph, but I think it's important that an app be able to access other people's everyone data.

(Example: If a non-friend comments on a friend's stream story, I should see their name/profile pic.)

3/ We need to let apps prompt users for their friends data. E.g., it *must* be possible that an app can ask to access your friends photos (e.g., a screensaver).

4/ 60 days is way too short. 4 months minimum. This will break every single application ever written. That said, we might say 60 days and then grant an extension.

**reply sent via email**

Mike Vernal - at 4:55pm on October 22nd
Mark asked me to add some important use cases for friend data.

1) Facebook for Adobe Air -- obviously you need access to the stream, friend (and non-friend) names, profile pictures, etc. Instead of "friend data", maybe the question should be something like "do you want to let this app see everything you can see on Facebook?" Or something that sound better than that.

2) Silverface -- this has a feature that lets you view all your friends' photos in a screensaver.

3) Birthday Reminders -- a bunch of apps help you keep track of your friends' birthdays.

Etc.

**reply sent via email**

Kelly Winters - at 5:50pm on October 22nd Here's a Google Doc I shared with a bunch more detail:

https://docs.google.com/a/facebook.com/Doc?docid=0AbFL56RvHZ3PZGc2Mmsza2JfN25naGs5NWQ1&hl=en

Use cases limited by losing non-public friend data Summary

```
  * "Not that private data"
      o No inviting based on stated interests (e.g., other friends that like chess)
      o Tell me who is having a good/bad day based on their status message
      o No more collaborative filtering (e.g., recommended books/music based on my friends' stated preferences)
          + Example: FriendRadio (http://www.facebook.com/friendradio) creates a radio station based on your friends' stated interests
      o No more online presence
          + Apps would not be able to tell a user that their friend is available to get engaged to use the app at the same time
  * "More private data"
      o No more dating apps (no relationship status)
      o No rich experiences based on detailed user data
          + Load viewing history of my friends' TV watching habits, recommend TV shows
  * "Lost aggregated viewing functionality"
      o No more content viewers/aggregators (notes, photos, etc.)
```

o Would this kill all desktop apps?

**reply sent via email**

Mark Kinsey - at 4:31pm on November 3rd
A few sketches of dialogs: This is the middle one - look at the two either side also:

http://www.intern.facebook.com/intern/pixelcloud/image.php?id=7691

Mark Kinsey - at 5:21pm on November 3rd
Here's a thought:

We should bucket the floodgate buckets (+ GDP specific ones), into 3/4 higher-level buckets, each of which has an nice graphic.

Developers request permissions at the floodgate bucket level, and an icon is shown for each higher-level bucket touched by this list.

This means we maintain the floodgate data bucketing for the developer interface, and use the higher-level stuff as just UI sugar.

Kelly Winters - at 6:33pm on November 3rd Cool dialogs. I think it'd also be powerful it showed actual data - e.g., my profile pic, my name, my email address. It's easy to ignore generic-looking text, but if you show me [redacted], I'm more likely to take notice.

Also - are we bucketing email in w/ other things vs. separating it out? Where's the best place to look at the intended buckets?

Mark Kinsey - at 7:31pm on November 3rd
JZ, Jackie and I had an impromptu chat, and have a solid plan for v1:

- Based on graphical buckets (similar to: http://www.intern.facebook.com/intern/pixelcloud/image.php?id=7692)

- Buckets would be:

1) Contact Info
2) Profile data
3) Friends data
4) Write to facebook
5) Manage Inbox

- Each bucket is represented by a suitable graphic on the left, and then the sub-permissions are enumerated to the right

- These are all the possible sub-permissions:
1) Contact info
  - Email (will need an elegant edit link)
  - Mailing address

2) Profile data (this is the same a floodgate buckets)
- About Me: detail1, detail2
- Friends and Relationships: detail1, detail2 etc
- Work and Education
- Content I Post
- Photos and Videos of Me
- Birthday
- Religious and Political

3) Friends data
- Use floodgate buckets tweaked as necessary

4) Write to facebook (use a sentence here as it's easier to explain)
- This application can automatically publish onto you facebook stream and your friends' facebook streams

5) Manage Inbox
- This application can manage your facebook inbox

- The dialog will be determined as follows:

1) The developer requests at a granular level what data they need (interests, birthday etc.)
2) We compare this to the user's everyone data and get a list of all things that are NOT everyone data
3) For every bucket touched by a list item, we spawn the graphic and enumerate the matching sub-permissions in the list

- My guess at the implications of this determination process:

30% of apps will show no dialog at all (because everyone data is sufficient for them) 30% of apps will just have the Profile data bucket 20% of apps will have Profile data + Write to facebook bucket 20% of apps will have more than this

Kelly Winters - at 10:57pm on November 3rd
Question: to what extent will a user be able to say, "No, I don't want to share that" but still be able to use a less-functional version of the app? Or, will these permissions be gating for engagement? If so, how will we message?

Mark Kinsey - at 4:15pm on November 4th
I think we should avoid making this stuff editable, the more I think about it. It makes the process more complex for users, and developers don't want to have to deal with users incoming with different amounts of data available.

So, yes - if you don't accept the upfront permissions/data, you can't use the app. I think the app could later ask for more permissions, for example...

I might be reluctant to give stream_publish to an app when I first use it (and the developer would know from their data that asking for it reduces conversions), so I don't get prompted. A few weeks later, I like the app and will accept the app's request for stream_publish.

Julie Zhuo - at 4:31pm on November 4th
Latest pixelcloud mock here:

http://www.intern.facebook.com/intern/pixelcloud/image.php?id=7707

Kelly Winters - at 4:37pm on November 4th Comments I added to the mock:

These are looking awesome. More questions:

1. Do users think about data + permissions as the same thing? Is it worth splitting them apart? (at the least, putting all "data" together and then "permissions" as a second bucket/section)
2. For "About Me" it'd be awesome if this included a preview of the data, e.g., "A limerick once asked a Joulee...", since I'm not sure all users immediately know what that includes. Same for birthday (e.g., do we pass birth year or not?)
3. My friends information - should it be "birthdays"?

Mike Vernal - at 5:10pm on November 4th
I really like the icons. :) I like the mocks and left a few comments there.

On the buckets -- I think the buckets defined by Mark in last post feel a little too coarse. What about something like the following?

* Profile Info (Silhouette Icon)
* Contact Info (Mail Message Icon)
* Photos (App Icon)
* Video (...)
* Notes
* Events
* Groups
* Mobile
* Etc.

Next to each bucket we can have a text description of what you're allowing. For apps, possibilities are read/write, i.e.:

* Access your photos
* Access your friends' photos
* Publish photos to Facebook
--
* Access your notes
* Access your friends' notes
* Publish notes to Facebook
--
Etc.

For Profile Info, I think we could use bullet points to spell out sub-profile buckets.

For Contact Info, we could use to say email vs. phone vs. address etc.

Thoughts? (I like this ...)

-mike

Mark Kinsey - at 5:14pm on November 4th
i feel pretty strongly about just have 4-5 high level buckets, each with an awesome chunky icon - such that user can calibrate quickly based on icons they recognize.

Mike Vernal - at 5:26pm on November 4th
I think what we should be maximizing for is fast user understanding.

I think you're asserting that 4-5 buckets is one way of accomplishing that.

I mostly agree (keeping buckets small), I'm just asserting that we should use concepts we already have rather than inventing new ones.

We already have these concepts on the site:
* Profile
* News Feed
* Photos
* Videos
* Events
* Groups
* Inbox
* Etc.

We should just use these concepts and their iconography to explain. So I agree w/ chunky buckets, I just don't want us to invent new ones that obscure what's going on.

If it makes my argument any strong, I too feel pretty strongly about this. ;)

-mike

Mark Kinsey - at 5:35pm on November 4th
nice comeback

i see your point, for sure. i'm concerned that with more buckets, we might feel better about the granularity we're giving users, but realistically they will not digest more than a few visual elements on a dialog.

let's chat in person tomorrow.

Mike Vernal - at 6:14pm on November 4th
Oh, I agree. IF there are more than 3-4 visual elements, people will be overwhelmed.

The key to keeping it simple, then, is making the profile/contact buckets small. That’s what I'm suggesting.

No app needs both photos + videos + notes + events + groups + etc. If there is one (e.g., a Facebook app that lets you do everything Facebook does), then maybe we can have one uber permission which is "let this app do everything in the world."

-mike

Kelly Winters - at 6:35pm on November 4th I kind of like how Julie's current mock handles this -- "Let this app communicate with the world for me via: Photos, Events OR Photos, Videos, OR Photos, Stream" etc. Be granular about how, but bucket them clearly.

Julie Zhuo - at 6:35pm on November 4th
I think one way to think about the buckets is that users should be able to make a gut decision based on the main idea of that bucket, and we should try to separate the buckets on what users would have varying levels of comfortableness around.

For example, if I grant access to my photos, I'm likely to have no problem granting access to notes, videos, events, groups. Hence, they should be grouped together. If I'm okay granting my friends' birthday, I'm probably also going to be okay with granting their favorite movies. Having more buckets when there isn't a real need for the distinction just ends up feeling more heavyweight.

I think we should try to avoid uber permissions if we can for the sake of consistency. If you are going to be a Facebook client app, I think the dialog should be pretty scary.

A good next step is to come up with actual scenarios for what type of apps need what data, and create the mocks based on those use cases.

Mike Vernal - at 6:44pm on November 4th
I'm actually not sure if this is the right way to think about it. It's not clear to me that granting access to photos means I want to grant access to notes/events/groups/etc.

If I'm using a photo screensaver app, I want it to ask for me+my friends profile info and me+my friends photos. If it's asking for notes or other random stuff, I'm going to think it's sketchy.

I agree about the uberpermission. It was premature optimization.

Mark Kinsey - at 1:46pm on November 5th
"If I'm using a photo screensaver app... If it's asking for notes or other random stuff, I'm going to think it's sketchy."

Exactly. We will list each thing it asking for next to the corresponding coarse icon. We're not suggesting hiding the granularity, just the level at which we use graphics.

Mark Kinsey - at 1:48pm on November 5th

Per JZs awesome point:

Kelly - would you mind putting together a list of say 20 apps which have these criteria:
1. We know the developers well enough to test stuff with them 2. In aggregate, are acceptably representative of the app community in terms of what data they ask (i.e. represent the spectrum, and balanced appropriately between lots and few data asks) 3. Includes canvas, connect and desktop

Then, we can mock up what the dialogs would look like for each.

Joe Sullivan - at 4:37pm on November 9th Hi -

I think we would want users to be able to select/opt out of sharing each different type of info -- Michael can confirm.

Joe

Kelly Winters - at 6:02pm on November 9th Hey guys,

Zhen added a bunch of sample partners here:
https://spreadsheets.google.com/a/facebook.com/ccc?key=t2E4qXxiItxb7pzpjr5PqJA&hl=en

LMK if you need more.

Kelly

Mike Vernal - at 6:18pm on November 9th
Joe/Michael --

While I understand that desire from user perspective, it doesn't make sense from a platform / developer perspective. If a permission is required, not providing it means the app will be broken. In that case, it is better to tell the user they can't use the app then to send them into a broken experience.

Mark Kinsey - at 6:25pm on November 9th
Zhen/Kelly - awesome. Can you clarify what cell E27 means, please?

Mike/all - Mike, Joe and I just chatted. Only optional data would be "opt out" (e.g. stuff the app might want for targeting). Mike's point still applies - having varying user data makes development troublesome. And it complicates things for users and us as well.

Also, we think the Canadians will be happy with no-edit. It would help to show: a) users don't care/use it, b) developers don't want it.

Kelly Winters - at 6:25pm on November 9th Have we considered testing the edit option? The privacy advocates I briefed for New Deal all asked about this, and generally prefer giving users options. If we can test it and show users don't use it, that will help bolster that argument.

Mark Kinsey - at 6:27pm on November 9th
we could just put an edit link at the bottom that goes nowhere and see if anyone clicks it!

Julie Zhuo - at 5:12pm on November 13th
Latest mock - http://www.intern.facebook.com/intern/pixelcloud/image.php?id=7810

Also, let's try to establish the buckets so we can get icons around each one (and determine ordering, which ones are particularly sensitive, etc.) Here is my first stab--feedback needed!

- Email
  - fields: Facebook contact email
  - icon: mailbox
- Personal Contact Info
  - fields: Address, Mobile/Land phone
  - icon: ?? house/phone?
  - sensitive
- SMS permissions
  - icon: mobile
  - sensitive
- About me
  - fields: About me, Personal info, website, hometown
  - icon: edit pencil
- Personal info about me
  - fields: birthday, religious political, family/relationship, education work
  - icon: silhouette man? ID card?
  - sensitive
- My content
  - fields: my stream, photos, videos, notes, events,
  - icon: applications icon?
  - sensitive
- Friends' info
  - fields: all floodgate buckets to which this applies
- Add content to Facebook
  - fields: stream.write, add photos, videos, notes
  - icon: arrow to news feed?
  - sensitive
- Manage Events (or should this go in the 'add content to fb bucket?)
  - fields: RSVP/create events
  - icon: events icon
  - sensitive
- Infinite Session
  - icon: ???
- Read Mailbox

- icon: ???

Mark Kinsey - at 5:42pm on November 16th Liking it. Thoughts:

- Need a snappier name for "Add Content'.. will come over to brainstorm sometime
- Infinite session: don't even show, because if we're removing caching policy it is moot
- 'Read Mailbox' should be 'Manage mailbox'
- Not sure if Events should go in 'Add Content...'

A few thoughts to test this framework. We don't want to be in a situation where we need to keep adding buckets.

What if we add? (possible bucket)
- Location (Personal info)
- Manage my Pages (new Manage Pages/Groups/Events bucket?)
- Manage my Groups (new Manage Pages/Groups/Events bucket?)
- Access my game scores (About me)

What about iphone/mobile dialogs?
- Phone numbers of my friends (Friend's info?)
- Chat history (About me)

Kelly Winters - at 10:31am on November 18th Few more questions on this:
Is there some kind of easy-to-follow framework we can use to make this readily clear to users? Read vs. write? Static vs. dynamic content? Me vs. Friends? Can we treat comm. Channels/ extended perms separately?

A few things it seemed like we might be missing:
- photos tagged of me -- devs have access to this, right? What about other things I'm tagged in?

Where would we add additional graph data into this?

Have we covered all data on this list? http://wiki.developers.facebook.com/index.php/FQL_Tables

Julie Zhuo - at 7:08pm on November 19th
Latest mock + buckets: http://www.intern.facebook.com/intern/pixelcloud/image.php?id=7882

Mark Kinsey - at 1:15pm
Quick check-in:
- We ran user tests last week (Jackie to provide update here)
- We had a few refinement to make based on these tests ( JZ/Jackie to do final iteration)
- Then we can hook it all up, start using it for extended permissions, plan the migration etc etc

Jackie Cerretani Frank - at 2:56pm
Thanks to the holiday or to general dereliction, only two of our users showed up. However, they provided some good insight into improving the design. We took them to three Connect sites (mocks), had them sign in using Connect, and go through the TOS, which varied in length and requests for each site. Here are the main takeaways: - The more information requested in the TOS, the more suspicious users were of sharing. This was successfully signaled by both the number of icons and the length of the list beneath each icon. - Users wanted more of an explanation of what the app planned to do with the information. - One user was completely confused, didn't understand the concept of connecting, and though all the categories were asking about permission to post back. This could have been related to confusion around the one-way arrow. - Giving away friends' information raised a red flag. They felt uncomfortable giving access, and confused about whether the friends' privacy settings would transfer to the Connect Site - In the event of a problem, users said they'd go to the Connect site first, FB second. - Users though the icons made sense and helped break up the information logically. Some users felt that categories could be collapsed, like email and contact info. - Users were confused by the yellow warning triangles, weren't sure of what they meant, and each user felt different things were most sensitive, not always overlapping with our designations. Suggestions generated in the debriefs: - We need to make the purpose of the dialog clearer. Consider having just one, bold line at the top with clearer language about what were asking for and what the benefit to the user is, with a more assuaging tone. - Consider trying alternate wording for buttons at the bottom to, again, give clues about what the dialog is asking for. - Consider giving more context about what is being shared inline under each bucket. *Concern that this will make the gray text artificially long, muddying the signal it's giving users about amount of info requested. -Try using a question mark icon link to give users a longer explanation of what's happening in dialog. - Find a way to make the arrow seem like it's going from all the icons to the site, or consider dropping the graphic altogether in favor of a list. - Consider nixing caution icons, since this is subjective and varies person to person. - Consider showing actual email address under the email icon - Consider showing what is actually being shared instead of the description. - Get feedback from devs by allowing them to mock up what they want to ask for and tell us how it works (since the launch isn't secret)

Jackie Cerretani Frank - at 3:04pm
Sorry that is so unreadable -- I can't get on the VPN from home today, so had to send by email. Will re-post a formatted version tomorrow. In the mentime, I'll email it to you guys in a seperate message.
Jackie

--------------------------------------------------------------------------------

Sent to subscribers: Amy Karasavas, Brent Goldman, Jackie Cerretani Frank, Jeffrey Spehar, Joe Sullivan, Julie Zhuo, Kelly Winters, Mark Kinsey, Matt Stone Trainer, Mike Vernal, Yariv Sadan